

LISTING OF THE CLAIMS

A copy of all pending claims and a status of the claims is provided below.

1. (Original) A method for optimizing pricing and capacity for bandwidth management using a computer, comprising the steps of:
 - inputting a mean and a variance of real usage for each of a plurality of customer classes;
 - inputting price and demand curve data which determines an arrival rate for each customer class;
 - inputting a number of existing customers in each customer class;
 - inputting a bandwidth wholesale cost;
 - generate a computer model for an optimization problem subject to a plurality of predetermined chance constraints;
 - solving said optimization problem using said computer to determine an amount of bandwidth to be purchased in a time period at a given price for an expected number of new customers in order to maximize profit; and
 - outputting said amount of bandwidth to be purchased and said expected number of new customers.

2. (Previously presented) A method for optimizing pricing and capacity for bandwidth management using a computer as recited in claim 1 wherein said plurality of predetermined chance constraints comprises:

$$b_{\tau} = b_{\tau-1} + a_{\tau} \quad (\tau = 1, \dots, T) \quad (1)$$

$$L_{i\tau} \leq q_{i\tau} \leq U_{i\tau} \quad (i = 1, \dots, I; \tau = 1, \dots, T) \quad (2)$$

$$\begin{aligned}
 & \sum_{i|\tau < d_i} [\lambda_{i\tau} \Delta \mu_i^2 + (n_{i\tau} + \lambda_{i\tau} \Delta)^2 \sigma_i^2 + (n_{i\tau} + \lambda_{i\tau} \Delta)^2 \mu_i^2] \\
 & + \sum_{i|\tau \geq d_i} [\lambda_i D_i (\mu_i^2 + \sigma_i^2)] + (\lambda_i D_i \mu_i)^2] - \delta_\tau b_\tau^2 \leq 0 \quad \forall \tau \quad (3)
 \end{aligned}$$

and said optimization problem comprises:

$$\text{Maximize}_{i,\tau} q_{i\tau} \lambda_i (q_{i\tau}) - \sum_{\tau} C_{\tau} a_{\tau} \quad (4)$$

wherein:

- $i = 1, \dots, I$: customer class;
- $\tau = 1, \dots, T$: time periods, each of length Δ ;
- δ_τ is tolerance on capacity violation in period τ ;
- C_{τ} is cost per unit of buying new capacity in period τ ;
- d_{τ} is duration of contract for customer class i ;
- D_i is actual duration of contract ($d_i \Delta$) for customer class i ;
- $n_{i\tau}$ is a number of existing contracts of type i still active at start of period τ ;
- $L_{i\tau}$ is a lower bound on contract price;
- $U_{i\tau}$ is an upper bound on contract price;
- b_{τ} is bandwidth available in period τ ,
- a_{τ} is bandwidth purchased by re-seller in period τ ;
- $q_{i\tau}$ is price to new or renewing customers for a new standard length contract of type i in period τ ;

$\lambda_i(q_{i\tau})$ is expected number of new customers of type i arriving in any period if a price for a contract is set at $q_{i\tau}$;

$\forall \tau$ is an index of a discrete time interval;

μ_i is mean time with customer class i ; and

δ_i is probability with customer class i .

3. (Original) A method for optimizing pricing and capacity for bandwidth management using a computer as recited in claim 1 wherein said computer solving and optimization problem is running a non-linear programming software.

4. (Original) A computer readable medium comprising software for causing a computer to execute steps for optimizing pricing and capacity for bandwidth management, comprising the steps of:

receiving a mean and a variance of real usage for each of a plurality of customer classes;

receiving price and demand curve data which determines an arrival rate for each customer class;

receiving a number of existing customers in each customer class;

receiving a bandwidth wholesale cost;

generating a computer model for an optimization problem subject to a plurality of predetermined chance constraints;

solving said optimization problem using said computer to determine an amount of bandwidth to be purchased in a time period at a given price for an expected number of new customers in order to maximize profit; and

outputting said amount of bandwidth to be purchased and said expected number of new customers.

5. (Previously presented) A computer readable medium comprising software for causing a computer to execute steps for optimizing pricing and capacity for bandwidth management as

recited in claim 4 wherein said plurality of predetermined chance constraints comprises:

$$b_\tau = b_{\tau-1} + a_\tau \quad (\tau = 1, \dots, T) \quad (1)$$

$$L_{i\tau} \leq q_{i\tau} \leq U_{i\tau} \quad (i = 1, \dots, I; \tau = 1, \dots, T) \quad (2)$$

$$\sum_{i|\tau < d_i} [\lambda_{i\tau} \Delta \mu_i^2 + (n_{i\tau} + \lambda_{i\tau} \Delta)^2 \sigma_i^2 + (n_{i\tau} + \lambda_{i\tau} \Delta)^2 \mu_i^2]$$

$$+ \sum_{i|\tau \geq d_i} [\lambda_i D_i (\mu_i^2 + \sigma_i^2)] + (\lambda_i D_i \mu_i)^2] - \delta_\tau b_\tau^2 \leq 0 \quad \forall \tau \quad (3)$$

and said optimization problem comprises:

$$\text{Maximize } \sum_{i,\tau} q_{i\tau} \lambda_i (q_{i\tau}) - \sum_\tau C_\tau a_\tau \quad (4)$$

wherein:

$i = 1, \dots, I$: customer class;

$\tau = 1, \dots, T$: time periods, each of length Δ ;

δ_τ is tolerance on capacity violation in period τ ;

C_τ is cost per unit of buying new capacity in period τ ;

d_τ is duration of contract for customer class i ;

D_i is actual duration of contract (d, Δ) for customer class i ;

$n_{i\tau}$ is number of existing contracts of type i still active at start of period τ ;

$L_{i\tau}$ is a lower bound on contract price;

$U_{i\tau}$ is an upper bound on contract price;

b_τ is bandwidth available in period τ ,

a_τ is bandwidth purchased by re-seller in period τ ;

$q_{i\tau}$ is price to new or renewing customers for a new standard length contract of type i in period τ ;

$\lambda_i(q_{i\tau})$ is expected number of new customers of type i arriving in any period if a price for a contract is set at $q_{i\tau}$;

$\forall \tau$ is an index of a discrete time interval;

μ_i is mean time with customer class i ; and

δ_i is probability with customer class i .

6. (Original) A computer readable medium comprising software for causing a computer to execute steps for optimizing pricing and capacity for bandwidth management as recited in claim 4 wherein said computer solving and optimization problem is running a non-linear programming software.